UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/803,675	03/18/2004	Tony Chau	CA920030070US1	5939

35525 7590 01/10/2008

IBM CORP (YA)
C/O YEE & ASSOCIATES PC
P.O. BOX 802333
DALLAS, TX 75380

EXAMINER	
SMITH, GARRETT A	

ART UNIT	PAPER NUMBER
2168	

MAIL DATE	DELIVERY MODE
01/10/2008	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/803,675	CHAU ET AL.	
	Examiner	Art Unit	
	Garrett A. Smith	2168	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *19 November 2007*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1 – 5, 7 – 15, 17 – 27 and 29* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1 – 5, 7 – 15, 17 – 27 and 29* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date *19 Novemebr 2007*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. This Office action is regarding Applicant's response filed 19 November 2007 to a prior Office action. Claims 1 – 5, 7 – 15, 17 – 27 and 29 are pending. Claims 6, 16, 28, and 30 – 34 are canceled. Claims 1, 7 – 11, 17 – 23, 26, 27 and 29 are amended.

2. This Office Action is a Second Action, Final Rejection.

Information Disclosure Statement

3. The Examiner has considered the Information Disclosure Statement filed 19 November 2007. A copy of the annotated IDS is enclosed with this Office Action.

Claim Objections

4. The Examiner notes that claim 33 was canceled. Therefore, the objection to claim 33 is **withdrawn**.

5. The Examiner notes that claim 9 recites "export generation code". The Examiner objects to Claim 7 due to this minor informality. The Examiner requests that this be changed to "generating step" or equivalent.

Response to Arguments

35 USC 112, Second Paragraph

6. Applicant's arguments (page 14), filed 19 November 2007, regarding the rejection under 35 USC 112, Second Paragraph of claims 6, 16 and 28 have been fully considered and are persuasive. Claims 6, 16 and 28 have been canceled. For these

reasons, the rejection under 35 USC 12, Second Paragraph of claims 6, 16 and 28 is **withdrawn**.

35 USC 101

7. Applicant's arguments (page 14 – 15) and amendments, filed 19 November 2007, regarding the rejection under 35 USC 101 of claims 1 – 6 and 10 – 16 have been fully considered and are persuasive. Claims 6 and 16 have been canceled. Applicant amended claim 1 to recite a "computer useable recordable data storage medium". For these reasons, the rejection under 35 USC 101 of claim 1 – 6 and 10 – 16 is **withdrawn**.

8. Applicant's arguments (page 14 – 15) and amendments, filed 19 November 2007, regarding the rejection under 35 USC 101 of claims 7 – 9 and 21 – 28 have been fully considered but are not persuasive. While Applicant has amended claim 7 (and dependants) from a software tool to a method claim, the method lacks a useful, concrete and tangible result. Each of the claimed steps does not require a result to be stored or displayed. The generating of graphical representations does not require the result be stored in memory or presented to a user. An explicit recitation of storing the generated graphical representations (for example, in memory) may overcome this rejection. Claims 21 – 28 suffer an equivalent deficiency and may be corrected in the same manner as claim 7. For these reasons, the rejection under 35 USC 101 of claim 7 – 9 and 21 – 28 is **maintained**.

9. Applicant's arguments (page 14 – 15) and amendments, filed 19 November 2007, regarding the rejection under 35 USC 101 of claims 17 – 20 have been fully considered but are not persuasive. Applicant has amended the claim to recite, "A computer system in combination with an export tool." It appears to the Examiner that Applicant intends to place the invention in the system statutory category. However, two valid interpretations can be made.

The first interpretation is that the system has no components as the export tool is not claimed as part of the system. Under this interpretation, the lack of components results in the claims not being in a statutory category because system claims must have a least one component.

The second interpretation is that the system does require the export tool. However, as the only component is a software tool, the entire system is considered software. Therefore, the claim still fails to fall within a statutory category.

The Examiner suggests that if Applicant wishes to properly claim a software tool in a system claim, Applicant may use the following template:

> A computer system for <rest of preamble>, the computer system comprising:
> A memory (or other hardware component)
> A software tool (for example, Applicant's export tool), wherein the software tool comprises: <list of features>

While this suggestion may overcome the current 35 USC 101 rejection, it should not be assumed that it will help overcome any other rejection provided by the Examiner. This suggestion is provided for informative purposes only, as Applicant and Applicant's Representative(s) is in the best position to form the instant claims to Applicant's

interests. Applicant is invited to call the Examiner if Applicant has any questions or concerns regarding this rejection.

For these reasons, the rejection under 35 USC 101 of claim 17 – 20 is **maintained**.

10. Applicant's arguments (page 14 – 15) and amendments, filed 19 November 2007, regarding the rejection under 35 USC 101 of claims 30 – 34 have been fully considered and are persuasive. Claims 30 – 34 have been canceled. For these reasons, the rejection under 35 USC 101 of claim 30 – 34 is **withdrawn**.

35 USC 103(a)

11. Applicant's arguments (page 15 – 18) and amendments, filed 19 November 2007, regarding the rejection under 35 USC 103(a) of claims 6, 16, 28, and 30 – 34 have been fully considered and are persuasive. Claims 6, 16, 28, and 30 – 34 are canceled. For these reasons, the rejection under 35 USC 103(a) of claims 6, 16, 28, and 30 – 34 is **withdrawn**.

12. Applicant's arguments (page 15 – 18) and amendments, filed 19 November 2007, regarding the rejection under 35 USC 103(a) of claims 1 – 3, 7, 8, 10 – 13, 17 – 19, 21 – 25 and 29 have been fully considered and are persuasive in part.

Applicant argues that "a set of features" is not shown either by Brisson or Collier et al. The Examiner would first like to note that "set of features" can comprise a single feature (see claim 3). The Examiner would like to further note that nowhere is a "feature" explicitly defined and therefore a broadest reasonable interpretation is

required. Next, the Examiner points to col 5, lines 44 – 60 and figures 6A – 6C which show various graphical representations of BNF in the form a RR diagrams, with labeled "features" as the paths shown in the RR diagrams.

Further, Applicant argues that the prior art does not teach or suggest "that the pattern mapping is defined relative to the structural-text based representations." The Examiner notes that the relevant section of claim 1 is "each feature in set of features having an **associated** pattern mapping defined **relative** to the structural text-based representations" (emphasis added). BNF-to-RR diagram conversion relies on matching and determining start points and end points. In other words, the Brisson system relies on a matching to determine terminal symbols and non-terminal symbols and this is a form of pattern matching (col 5, lines 44 – 60).

Applicant takes issue with Examiner's statement "it is inherent in Brisson to convert a RR diagram into a BNF using pattern mapping." The Examiner agrees that Brisson does not disclose the reverse of changing a RR diagram into BNF notation and that this feature is not inherent in the reference. However, the Examiner notes that this is very well known in the art and taught by Lämmel et al ("Semi-automatic Grammar Recovery; available July 2001). In section 3.1 and 3.2, Lämmel et al discusses extraction of whole syntax (or railroad) diagrams from IBM's VS COBOL II manual and the parsing of the diagrams into BNF code. Therefore, a rejection under 35 USC 103(a) is provided below which includes this reference.

For these reasons, the rejection under 35 USC 103(a) of claims 1 – 3, 7, 8, 10 – 13, 17 – 19, 21 – 25 and 29 is **withdrawn**.

Also for these reasons, the rejection under 35 USC 103(a) of claims 4 – 6, 9, 14 – 16, 20 and 26 – 28 is **withdrawn**.

Claim Rejections - 35 USC § 112, First Paragraph

13. The following is a quotation of the first paragraph of 35 U.S.C. 112:

> The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

14. Claim **29** rejected under 35 U.S.C. 112, first paragraph, because the specification, while being enabling for accomplishing the method of claim 23, does not reasonably provide enablement for every possible combination that could accomplish the method of claim 23. See *In re Hyatt*, 708 F.2d 712, 714-715, 218 USPQ 195, 197 (Fed. Cir. 1983). *In re Hyatt* held that a single means claim which covered every conceivable means for achieving the stated purpose was nonenabling for the scope of the claim because the specification disclosed at most only known to the inventor (MPEP § 2164.08(a). In claim, Applicant appears to be attempting to claim every conceivable means for executing the steps of claim 23. As such, the specification does not enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make the invention commensurate in scope with these claims.

Claim Rejections - 35 USC § 101

15. 35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

16. Claims **7 – 9 and 17 – 28** are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter.

17. **Claims 7 – 9 and 21 – 28** do not provide for a useful concrete and tangible result. While Applicant has amended claim 7 (and dependants) from a software tool to a method claim, the method lacks a useful, concrete and tangible result. Each of the claimed steps does not require a result to be stored or displayed. The generating of graphical representations does not require the result be stored in memory or presented to a user. An explicit recitation of storing the generated graphical representations (for example, in memory) may overcome this rejection. Claims 21 – 28 suffer an equivalent deficiency and may be corrected in the same manner as claim 7.

18. **Claims 17 – 20** is directed toward nonstatutory subject matter. Applicant has amended the claim to recite, "A computer system in combination with an export tool." It appears to the Examiner that Applicant intends to place the invention in the system statutory category. However, two valid interpretations can be made.

The first interpretation is that the system has no components as the export tool is not claimed as part of the system. Under this interpretation, the lack of components results in the claims not being in a statutory category because system claims must have a least one component.

The second interpretation is that the system does require the export tool. However, as the only component is a software tool, the entire system is considered software. Therefore, the claim still fails to fall within a statutory category.

The Examiner suggests that if Applicant wishes to properly claim a software tool in a system claim, Applicant may use the following template:

> A computer system for <rest of preamble>, the computer system comprising:
> A memory (or other hardware component)
> A software tool (for example, Applicant's export tool), wherein the software tool comprises: <list of features>

While this suggestion may overcome the current 35 USC 101 rejection, it should not be assumed that it will help overcome any other rejection provided by the Examiner. This suggestion is provided for informative purposes only, as Applicant and Applicant's Representative(s) is in the best position to form the instant claims to Applicant's interests. Applicant is invited to call the Examiner if Applicant has any questions or concerns regarding this rejection.

Claim Rejections - 35 USC § 103

19. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

20. This application currently names joint inventors. In considering patentability of the claims under 35 U.S.C. 103(a), the examiner presumes that the subject matter of

the various claims was commonly owned at the time any inventions covered therein were made absent any evidence to the contrary. Applicant is advised of the obligation under 37 CFR 1.56 to point out the inventor and invention dates of each claim that was not commonly owned at the time a later invention was made in order for the examiner to consider the applicability of 35 U.S.C. 103(c) and potential 35 U.S.C. 102(e), (f) or (g) prior art under 35 U.S.C. 103(a).

21. **Claims 1 – 3, 7, 8, 10 – 13, 17 – 19, 21 – 25 and 29** are rejected under 35 U.S.C. 103(a) as being unpatentable over Brisson (US Patent 5,678,052; patented 14 October 1997) in view of Lämmel et al ("Semi-automatic Grammar Recovery; available July 2001) and further in view of Collier et al (US Patent 5,815,152).

22. In regard to **claim 1**, Brisson teaches methods and systems for converting a Backus-Naur Form (BNF) grammar (structural text-based) into a compressed rail-road (RR) diagram (see Abstract, as well as Figure 5 for the example of a BNF and Figure 6a, 6b, 6c for the RR diagram of that BNF). Brisson discloses using pattern mappings to create the RR diagram (see figure 3).

While Brisson does not directly disclose creating a BNF from a RR diagram, Lämmel et al discusses in section 3.1 and 3.2 extraction of whole syntax (i.e. railroad) diagrams from IBM's VS COBOL II manual and the parsing of the diagrams into BNF code. It would have been obvious to a person of ordinary skill at the time of invention to use a the parser of Lämmel et al with the system of Brisson because it provides for a user to check and verify the validity of a provide BNF or RR diagram. Further, it allows

a user to quickly convert a RR diagram into a machine readable form for processing by a computer without having to manually convert it (section 3.1 of Lämmel et al).

Brisson does not explicitly teach that the representations can be business processes. Collier et al does teach that graphical representation of business processes can be created through an interface (see figure 5). It would have been obvious to a person of ordinary skill at the time of invention to use the conversion methods and systems of Brisson and Lämmel et al with the graphical business process representations of Collier et al because it would resulted in the predictable result of converting text-based business processes into graphical representations of the business processes and furthermore would allow a user to have a multi-format business process that could be used elsewhere.

23. In regard to **claim 2**, Collier et al further teaches a system that allows a user to construct a business process with nodes that include conditionals, send/receive/response actions, and iterations (see figures 5, 8 and 11).

While Brisson does not directly disclose creating a BNF from a RR diagram, Lämmel et al discusses in section 3.1 and 3.2 extraction of whole syntax (i.e. railroad) diagrams from IBM's VS COBOL II manual and the parsing of the diagrams into BNF code. It would have been obvious to a person of ordinary skill at the time of invention to use a the parser of Lämmel et al with the system of Brisson because it provides for a user to check and verify the validity of a provide BNF or RR diagram. Further, it allows a user to quickly convert a RR diagram into a machine readable form for processing by a computer without having to manually convert it (section 3.1 of Lämmel et al).

Brisson does not explicitly teach that the representations can be business processes. Collier et al does teach that graphical representation of business processes can be created through an interface (see figure 5). It would have been obvious to a person of ordinary skill at the time of invention to use the conversion methods and systems of Brisson and Lämmel et al with the graphical business process representations of Collier et al because it would resulted in the predictable result of converting text-based business processes into graphical representations of the business processes and furthermore would allow a user to have a multi-format business process that could be used elsewhere.

24. In regard to **claim 3**, Brisson, as mentioned above, teaches pattern mapping for converting between a BNF and a RR diagram (col 5, lines 44 – 60). Collier et al further teaches a system that allows a user to construct a business process with nodes that include conditionals, send/receive/response actions, and iterations (see figures 5, 8 and 11).

While Brisson does not directly disclose creating a BNF from a RR diagram, Lämmel et al discusses in section 3.1 and 3.2 extraction of whole syntax (i.e. railroad) diagrams from IBM's VS COBOL II manual and the parsing of the diagrams into BNF code. It would have been obvious to a person of ordinary skill at the time of invention to use a the parser of Lämmel et al with the system of Brisson because it provides for a user to check and verify the validity of a provide BNF or RR diagram. Further, it allows a user to quickly convert a RR diagram into a machine readable form for processing by a computer without having to manually convert it (section 3.1 of Lämmel et al).

Brisson does not explicitly teach that the representations can be business processes. Collier et al does teach that graphical representation of business processes can be created through an interface (see figure 5). It would have been obvious to a person of ordinary skill at the time of invention to use the conversion methods and systems of Brisson and Lämmel et al with the graphical business process representations of Collier et al because it would resulted in the predictable result of converting text-based business processes into graphical representations of the business processes and furthermore would allow a user to have a multi-format business process that could be used elsewhere.

25. In regard to **claim 7**, Brisson teaches methods and systems for converting a Backus-Naur Form (BNF) grammar (structural text-based) into a compressed rail-road (RR) diagram (see Abstract, as well as Figure 5 for the example of a BNF and Figure 6a, 6b, 6c for the RR diagram of that BNF). Brisson discloses using pattern mappings to create the RR diagram (see figure 3).

While Brisson does not directly disclose creating a BNF from a RR diagram, Lämmel et al discusses in section 3.1 and 3.2 extraction of whole syntax (i.e. railroad) diagrams from IBM's VS COBOL II manual and the parsing of the diagrams into BNF code. It would have been obvious to a person of ordinary skill at the time of invention to use a the parser of Lämmel et al with the system of Brisson because it provides for a user to check and verify the validity of a provide BNF or RR diagram. Further, it allows a user to quickly convert a RR diagram into a machine readable form for processing by a computer without having to manually convert it (section 3.1 of Lämmel et al).

Brisson does not explicitly teach that the representations can be business processes. Collier et al does teach that graphical representation of business processes can be created through an interface (see figure 5). It would have been obvious to a person of ordinary skill at the time of invention to use the conversion methods and systems of Brisson and Lämmel et al with the graphical business process representations of Collier et al because it would resulted in the predictable result of converting text-based business processes into graphical representations of the business processes and furthermore would allow a user to have a multi-format business process that could be used elsewhere.

26. In regard to **claim 8**, Brisson, as mentioned above, teaches pattern mapping for converting between a BNF and a RR diagram (col 5, lines 44 – 60). Collier et al further teaches a system that allows a user to construct a business process with nodes that include conditionals, send/receive/response actions, and iterations (see figures 5, 8 and 11).

While Brisson does not directly disclose creating a BNF from a RR diagram, Lämmel et al discusses in section 3.1 and 3.2 extraction of whole syntax (i.e. railroad) diagrams from IBM's VS COBOL II manual and the parsing of the diagrams into BNF code. It would have been obvious to a person of ordinary skill at the time of invention to use a the parser of Lämmel et al with the system of Brisson because it provides for a user to check and verify the validity of a provide BNF or RR diagram. Further, it allows a user to quickly convert a RR diagram into a machine readable form for processing by a computer without having to manually convert it (section 3.1 of Lämmel et al).

Brisson does not explicitly teach that the representations can be business processes. Collier et al does teach that graphical representation of business processes can be created through an interface (see figure 5). It would have been obvious to a person of ordinary skill at the time of invention to use the conversion methods and systems of Brisson and Lämmel et al with the graphical business process representations of Collier et al because it would resulted in the predictable result of converting text-based business processes into graphical representations of the business processes and furthermore would allow a user to have a multi-format business process that could be used elsewhere.

27. In regard to **claim 10**, Brisson teaches methods and systems for converting a Backus-Naur Form (BNF) grammar (structural text-based) into a compressed rail-road (RR) diagram (see Abstract, as well as Figure 5 for the example of a BNF and Figure 6a, 6b, 6c for the RR diagram of that BNF). Brisson discloses using pattern mappings to create the RR diagram (see figure 3).

While Brisson does not directly disclose creating a BNF from a RR diagram, Lämmel et al discusses in section 3.1 and 3.2 extraction of whole syntax (i.e. railroad) diagrams from IBM's VS COBOL II manual and the parsing of the diagrams into BNF code. It would have been obvious to a person of ordinary skill at the time of invention to use a the parser of Lämmel et al with the system of Brisson because it provides for a user to check and verify the validity of a provide BNF or RR diagram. Further, it allows a user to quickly convert a RR diagram into a machine readable form for processing by a computer without having to manually convert it (section 3.1 of Lämmel et al).

Brisson does not explicitly teach that the representations can be business processes. Collier et al does teach that graphical representation of business processes can be created through an interface (see figure 5). It would have been obvious to a person of ordinary skill at the time of invention to use the conversion methods and systems of Brisson and Lämmel et al with the graphical business process representations of Collier et al because it would resulted in the predictable result of converting text-based business processes into graphical representations of the business processes and furthermore would allow a user to have a multi-format business process that could be used elsewhere.

28. In regard to **claim 11**, Brisson teaches methods and systems for converting a Backus-Naur Form (BNF) grammar (structural text-based) into a compressed rail-road (RR) diagram (see Abstract, as well as Figure 5 for the example of a BNF and Figure 6a, 6b, 6c for the RR diagram of that BNF). Brisson discloses using pattern mappings to create the RR diagram (see figure 3).

While Brisson does not directly disclose creating a BNF from a RR diagram, Lämmel et al discusses in section 3.1 and 3.2 extraction of whole syntax (i.e. railroad) diagrams from IBM's VS COBOL II manual and the parsing of the diagrams into BNF code. It would have been obvious to a person of ordinary skill at the time of invention to use a the parser of Lämmel et al with the system of Brisson because it provides for a user to check and verify the validity of a provide BNF or RR diagram. Further, it allows a user to quickly convert a RR diagram into a machine readable form for processing by a computer without having to manually convert it (section 3.1 of Lämmel et al).

Brisson does not explicitly teach that the representations can be business processes. Collier et al does teach that graphical representation of business processes can be created through an interface (see figure 5). It would have been obvious to a person of ordinary skill at the time of invention to use the conversion methods and systems of Brisson and Lämmel et al with the graphical business process representations of Collier et al because it would resulted in the predictable result of converting text-based business processes into graphical representations of the business processes and furthermore would allow a user to have a multi-format business process that could be used elsewhere.

29. In regard to **claim 12**, Collier et al further teaches a system that allows a user to construct a business process with nodes that include conditionals, send/receive/response actions, and iterations (see figures 5, 8 and 11).

While Brisson does not directly disclose creating a BNF from a RR diagram, Lämmel et al discusses in section 3.1 and 3.2 extraction of whole syntax (i.e. railroad) diagrams from IBM's VS COBOL II manual and the parsing of the diagrams into BNF code. It would have been obvious to a person of ordinary skill at the time of invention to use a the parser of Lämmel et al with the system of Brisson because it provides for a user to check and verify the validity of a provide BNF or RR diagram. Further, it allows a user to quickly convert a RR diagram into a machine readable form for processing by a computer without having to manually convert it (section 3.1 of Lämmel et al).

Brisson does not explicitly teach that the representations can be business processes. Collier et al does teach that graphical representation of business processes

can be created through an interface (see figure 5). It would have been obvious to a person of ordinary skill at the time of invention to use the conversion methods and systems of Brisson and Lämmel et al with the graphical business process representations of Collier et al because it would resulted in the predictable result of converting text-based business processes into graphical representations of the business processes and furthermore would allow a user to have a multi-format business process that could be used elsewhere.

30. In regard to **claim 13**, Brisson, as mentioned above, teaches pattern mapping for converting between a BNF and a RR diagram (col 5, lines 44 – 60). Collier et al further teaches a system that allows a user to construct a business process with nodes that include conditionals, send/receive/response actions, and iterations (see figures 5, 8 and 11).

While Brisson does not directly disclose creating a BNF from a RR diagram, Lämmel et al discusses in section 3.1 and 3.2 extraction of whole syntax (i.e. railroad) diagrams from IBM's VS COBOL II manual and the parsing of the diagrams into BNF code. It would have been obvious to a person of ordinary skill at the time of invention to use a the parser of Lämmel et al with the system of Brisson because it provides for a user to check and verify the validity of a provide BNF or RR diagram. Further, it allows a user to quickly convert a RR diagram into a machine readable form for processing by a computer without having to manually convert it (section 3.1 of Lämmel et al).

Brisson does not explicitly teach that the representations can be business processes. Collier et al does teach that graphical representation of business processes

can be created through an interface (see figure 5). It would have been obvious to a person of ordinary skill at the time of invention to use the conversion methods and systems of Brisson and Lämmel et al with the graphical business process representations of Collier et al because it would resulted in the predictable result of converting text-based business processes into graphical representations of the business processes and furthermore would allow a user to have a multi-format business process that could be used elsewhere.

31. In regard to **claim 17**, Brisson teaches methods and systems for converting a Backus-Naur Form (BNF) grammar (structural text-based) into a compressed rail-road (RR) diagram (see Abstract, as well as Figure 5 for the example of a BNF and Figure 6a, 6b, 6c for the RR diagram of that BNF). Brisson discloses using pattern mappings to create the RR diagram (see figure 3).

While Brisson does not directly disclose creating a BNF from a RR diagram, Lämmel et al discusses in section 3.1 and 3.2 extraction of whole syntax (i.e. railroad) diagrams from IBM's VS COBOL II manual and the parsing of the diagrams into BNF code. It would have been obvious to a person of ordinary skill at the time of invention to use a the parser of Lämmel et al with the system of Brisson because it provides for a user to check and verify the validity of a provide BNF or RR diagram. Further, it allows a user to quickly convert a RR diagram into a machine readable form for processing by a computer without having to manually convert it (section 3.1 of Lämmel et al).

Brisson does not explicitly teach that the representations can be business processes. Collier et al does teach that graphical representation of business processes

can be created through an interface (see figure 5). It would have been obvious to a person of ordinary skill at the time of invention to use the conversion methods and systems of Brisson and Lämmel et al with the graphical business process representations of Collier et al because it would resulted in the predictable result of converting text-based business processes into graphical representations of the business processes and furthermore would allow a user to have a multi-format business process that could be used elsewhere.

32. In regard to **claim 18**, Collier et al further teaches a system that allows a user to construct a business process with nodes that include conditionals, send/receive/response actions, and iterations (see figures 5, 8 and 11).

While Brisson does not directly disclose creating a BNF from a RR diagram, Lämmel et al discusses in section 3.1 and 3.2 extraction of whole syntax (i.e. railroad) diagrams from IBM's VS COBOL II manual and the parsing of the diagrams into BNF code. It would have been obvious to a person of ordinary skill at the time of invention to use a the parser of Lämmel et al with the system of Brisson because it provides for a user to check and verify the validity of a provide BNF or RR diagram. Further, it allows a user to quickly convert a RR diagram into a machine readable form for processing by a computer without having to manually convert it (section 3.1 of Lämmel et al).

Brisson does not explicitly teach that the representations can be business processes. Collier et al does teach that graphical representation of business processes can be created through an interface (see figure 5). It would have been obvious to a person of ordinary skill at the time of invention to use the conversion methods and

systems of Brisson and Lämmel et al with the graphical business process representations of Collier et al because it would resulted in the predictable result of converting text-based business processes into graphical representations of the business processes and furthermore would allow a user to have a multi-format business process that could be used elsewhere.

33. In regard to **claim 19**, Brisson, as mentioned above, teaches pattern mapping for converting between a BNF and a RR diagram (col 5, lines 44 – 60). Collier et al further teaches a system that allows a user to construct a business process with nodes that include conditionals, send/receive/response actions, and iterations (see figures 5, 8 and 11).

While Brisson does not directly disclose creating a BNF from a RR diagram, Lämmel et al discusses in section 3.1 and 3.2 extraction of whole syntax (i.e. railroad) diagrams from IBM's VS COBOL II manual and the parsing of the diagrams into BNF code. It would have been obvious to a person of ordinary skill at the time of invention to use a the parser of Lämmel et al with the system of Brisson because it provides for a user to check and verify the validity of a provide BNF or RR diagram. Further, it allows a user to quickly convert a RR diagram into a machine readable form for processing by a computer without having to manually convert it (section 3.1 of Lämmel et al).

Brisson does not explicitly teach that the representations can be business processes. Collier et al does teach that graphical representation of business processes can be created through an interface (see figure 5). It would have been obvious to a person of ordinary skill at the time of invention to use the conversion methods and

systems of Brisson and Lämmel et al with the graphical business process representations of Collier et al because it would resulted in the predictable result of converting text-based business processes into graphical representations of the business processes and furthermore would allow a user to have a multi-format business process that could be used elsewhere.

34. In regard to **claim 21**, Brisson teaches methods and systems for converting a Backus-Naur Form (BNF) grammar (structural text-based) into a compressed rail-road (RR) diagram (see Abstract, as well as Figure 5 for the example of a BNF and Figure 6a, 6b, 6c for the RR diagram of that BNF). Brisson discloses using pattern mappings to create the RR diagram (see figure 3).

While Brisson does not directly disclose creating a BNF from a RR diagram, Lämmel et al discusses in section 3.1 and 3.2 extraction of whole syntax (i.e. railroad) diagrams from IBM's VS COBOL II manual and the parsing of the diagrams into BNF code. It would have been obvious to a person of ordinary skill at the time of invention to use a the parser of Lämmel et al with the system of Brisson because it provides for a user to check and verify the validity of a provide BNF or RR diagram. Further, it allows a user to quickly convert a RR diagram into a machine readable form for processing by a computer without having to manually convert it (section 3.1 of Lämmel et al).

Brisson does not explicitly teach that the representations can be business processes. Collier et al does teach that graphical representation of business processes can be created through an interface (see figure 5). It would have been obvious to a person of ordinary skill at the time of invention to use the conversion methods and

systems of Brisson and Lämmel et al with the graphical business process representations of Collier et al because it would resulted in the predictable result of converting text-based business processes into graphical representations of the business processes and furthermore would allow a user to have a multi-format business process that could be used elsewhere.

35. In regard to **claim 22**, Brisson, as mentioned above, teaches pattern mapping for converting between a BNF and a RR diagram (col 5, lines 44 – 60). Collier et al further teaches a system that allows a user to construct a business process with nodes that include conditionals, send/receive/response actions, and iterations (see figures 5, 8 and 11).

While Brisson does not directly disclose creating a BNF from a RR diagram, Lämmel et al discusses in section 3.1 and 3.2 extraction of whole syntax (i.e. railroad) diagrams from IBM's VS COBOL II manual and the parsing of the diagrams into BNF code. It would have been obvious to a person of ordinary skill at the time of invention to use a the parser of Lämmel et al with the system of Brisson because it provides for a user to check and verify the validity of a provide BNF or RR diagram. Further, it allows a user to quickly convert a RR diagram into a machine readable form for processing by a computer without having to manually convert it (section 3.1 of Lämmel et al).

Brisson does not explicitly teach that the representations can be business processes. Collier et al does teach that graphical representation of business processes can be created through an interface (see figure 5). It would have been obvious to a person of ordinary skill at the time of invention to use the conversion methods and

systems of Brisson and Lämmel et al with the graphical business process representations of Collier et al because it would resulted in the predictable result of converting text-based business processes into graphical representations of the business processes and furthermore would allow a user to have a multi-format business process that could be used elsewhere.

36. In regard to **claim 23**, Brisson teaches methods and systems for converting a Backus-Naur Form (BNF) grammar (structural text-based) into a compressed rail-road (RR) diagram (see Abstract, as well as Figure 5 for the example of a BNF and Figure 6a, 6b, 6c for the RR diagram of that BNF). Brisson discloses using pattern mappings to create the RR diagram (see figure 3).

While Brisson does not directly disclose creating a BNF from a RR diagram, Lämmel et al discusses in section 3.1 and 3.2 extraction of whole syntax (i.e. railroad) diagrams from IBM's VS COBOL II manual and the parsing of the diagrams into BNF code. It would have been obvious to a person of ordinary skill at the time of invention to use a the parser of Lämmel et al with the system of Brisson because it provides for a user to check and verify the validity of a provide BNF or RR diagram. Further, it allows a user to quickly convert a RR diagram into a machine readable form for processing by a computer without having to manually convert it (section 3.1 of Lämmel et al).

Brisson does not explicitly teach that the representations can be business processes. Collier et al does teach that graphical representation of business processes can be created through an interface (see figure 5). It would have been obvious to a person of ordinary skill at the time of invention to use the conversion methods and

systems of Brisson and Lämmel et al with the graphical business process representations of Collier et al because it would resulted in the predictable result of converting text-based business processes into graphical representations of the business processes and furthermore would allow a user to have a multi-format business process that could be used elsewhere.

37. In regard to **claim 24**, Collier et al further teaches a system that allows a user to construct a business process with nodes that include conditionals, send/receive/response actions, and iterations (see figures 5, 8 and 11).

While Brisson does not directly disclose creating a BNF from a RR diagram, Lämmel et al discusses in section 3.1 and 3.2 extraction of whole syntax (i.e. railroad) diagrams from IBM's VS COBOL II manual and the parsing of the diagrams into BNF code. It would have been obvious to a person of ordinary skill at the time of invention to use a the parser of Lämmel et al with the system of Brisson because it provides for a user to check and verify the validity of a provide BNF or RR diagram. Further, it allows a user to quickly convert a RR diagram into a machine readable form for processing by a computer without having to manually convert it (section 3.1 of Lämmel et al).

Brisson does not explicitly teach that the representations can be business processes. Collier et al does teach that graphical representation of business processes can be created through an interface (see figure 5). It would have been obvious to a person of ordinary skill at the time of invention to use the conversion methods and systems of Brisson and Lämmel et al with the graphical business process representations of Collier et al because it would resulted in the predictable result of

converting text-based business processes into graphical representations of the business processes and furthermore would allow a user to have a multi-format business process that could be used elsewhere.

38. In regard to **claim 25**, Brisson, as mentioned above, teaches pattern mapping for converting between a BNF and a RR diagram (col 5, lines 44 – 60). Collier et al further teaches a system that allows a user to construct a business process with nodes that include conditionals, send/receive/response actions, and iterations (see figures 5, 8 and 11).

While Brisson does not directly disclose creating a BNF from a RR diagram, Lämmel et al discusses in section 3.1 and 3.2 extraction of whole syntax (i.e. railroad) diagrams from IBM's VS COBOL II manual and the parsing of the diagrams into BNF code. It would have been obvious to a person of ordinary skill at the time of invention to use a the parser of Lämmel et al with the system of Brisson because it provides for a user to check and verify the validity of a provide BNF or RR diagram. Further, it allows a user to quickly convert a RR diagram into a machine readable form for processing by a computer without having to manually convert it (section 3.1 of Lämmel et al).

Brisson does not explicitly teach that the representations can be business processes. Collier et al does teach that graphical representation of business processes can be created through an interface (see figure 5). It would have been obvious to a person of ordinary skill at the time of invention to use the conversion methods and systems of Brisson and Lämmel et al with the graphical business process representations of Collier et al because it would resulted in the predictable result of

converting text-based business processes into graphical representations of the business processes and furthermore would allow a user to have a multi-format business process that could be used elsewhere.

In regard to **claim 29**, Brisson teaches a recordable data storage medium in figure 1. While Brisson does not directly disclose creating a BNF from a RR diagram, Lämmel et al discusses in section 3.1 and 3.2 extraction of whole syntax (i.e. railroad) diagrams from IBM's VS COBOL II manual and the parsing of the diagrams into BNF code. It would have been obvious to a person of ordinary skill at the time of invention to use a the parser of Lämmel et al with the system of Brisson because it provides for a user to check and verify the validity of a provide BNF or RR diagram. Further, it allows a user to quickly convert a RR diagram into a machine readable form for processing by a computer without having to manually convert it (section 3.1 of Lämmel et al).

Brisson does not explicitly teach that the representations can be business processes. Collier et al does teach that graphical representation of business processes can be created through an interface (see figure 5). It would have been obvious to a person of ordinary skill at the time of invention to use the conversion methods and systems of Brisson and Lämmel et al with the graphical business process representations of Collier et al because it would resulted in the predictable result of converting text-based business processes into graphical representations of the business processes and furthermore would allow a user to have a multi-format business process that could be used elsewhere.

39. **Claims 4, 5, 9, 14, 15, 20, 26 and 27** are rejected under 35 U.S.C. 103(a) as being unpatentable over Brisson (US Patent 5,678,052; patented 14 October 1997) in view of Lämmel et al ("Semi-automatic Grammar Recovery; available July 2001) and Collier et al (US Patent 5,815,152) as applied above and in further view of Nemer (US PGPUB 2003/0110446, published 12 June 2003).

40. In regard to **claim 4**, Nemer teaches an object class and method for converting XML (which XPath is part of) to Java code and Java into XML code (see figure 3 and 4). It would have been obvious to a person of ordinary skill in the at the time of invention to use the conversion system of Nemer with the system of Brisson and Collier et al because it would have given the predictable result of converting a structured text based language (XML) into a code that can be used for graphical representations (Java).

41. In regard to **claim 5**, Nemer teaches an object class and method for converting XML (which XPath is part of) to Java code and Java into XML code (see figure 3 and 4). It would have been obvious to a person of ordinary skill in the at the time of invention to use the conversion system of Nemer with the system of Brisson and Collier et al because it would have given the predictable result of converting a structured text based language (XML) into a code that can be used for graphical representations (Java).

42. In regard to **claim 9**, Nemer teaches an object class and method for converting XML (which XPath is part of) to Java code and Java into XML code (see figure 3 and 4). It would have been obvious to a person of ordinary skill in the at the time of invention to use the conversion system of Nemer with the system of Brisson and Collier et al

because it would have given the predictable result of converting a structured text based language (XML) into a code that can be used for graphical representations (Java).

43. In regard to **claim 14**, Nemer teaches an object class and method for converting XML (which XPath is part of) to Java code and Java into XML code (see figure 3 and 4). It would have been obvious to a person of ordinary skill in the at the time of invention to use the conversion system of Nemer with the system of Brisson and Collier et al because it would have given the predictable result of converting a structured text based language (XML) into a code that can be used for graphical representations (Java).

44. In regard to **claim 15**, Nemer teaches an object class and method for converting XML (which XPath is part of) to Java code and Java into XML code (see figure 3 and 4). It would have been obvious to a person of ordinary skill in the at the time of invention to use the conversion system of Nemer with the system of Brisson and Collier et al because it would have given the predictable result of converting a structured text based language (XML) into a code that can be used for graphical representations (Java).

45. In regard to **claim 20**, Nemer teaches an object class and method for converting XML (which XPath is part of) to Java code and Java into XML code (see figure 3 and 4). It would have been obvious to a person of ordinary skill in the at the time of invention to use the conversion system of Nemer with the system of Brisson and Collier et al because it would have given the predictable result of converting a structured text based language (XML) into a code that can be used for graphical representations (Java).

46. In regard to **claim 26**, Nemer teaches an object class and method for converting XML (which XPath is part of) to Java code and Java into XML code (see figure 3 and 4).

It would have been obvious to a person of ordinary skill in the at the time of invention to use the conversion system of Nemer with the system of Brisson and Collier et.al because it would have given the predictable result of converting a structured text based language (XML) into a code that can be used for graphical representations (Java).

47. In regard to **claim 27**, Nemer teaches an object class and method for converting XML (which XPath is part of) to Java code and Java into XML code (see figure 3 and 4). It would have been obvious to a person of ordinary skill in the at the time of invention to use the conversion system of Nemer with the system of Brisson and Collier et al because it would have given the predictable result of converting a structured text based language (XML) into a code that can be used for graphical representations (Java).

Conclusion

48. The Examiner requests, in response to this Office action, that support be shown for language added to any original claims on amendment and any new claims. That is, indicate support for newly added claim language by specifically pointing to page(s) and line no(s) in the specification and/or drawing figure(s). This will assist the Examiner in prosecuting the application.

49. When responding to this Office action, Applicant is advised to clearly point out the patentable novelty which he or she thinks the claims present, in view of the state of the art disclosed by the references cited or the objections made. He or she must also show how the amendments avoid such references or objections See 37 CFR 1.111(c).

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Garrett A. Smith whose telephone number is (571) 270-1764. If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tim T. Vo can be reached on (571) 272-3642. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

January 5, 2008

Garrett Smith
Patent Examiner
Art Unit 2168

TIM VO
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100